This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

As rescanning documents *will not* correct images, please do not report the images to the Image Problem Mailbox.

Fig.1

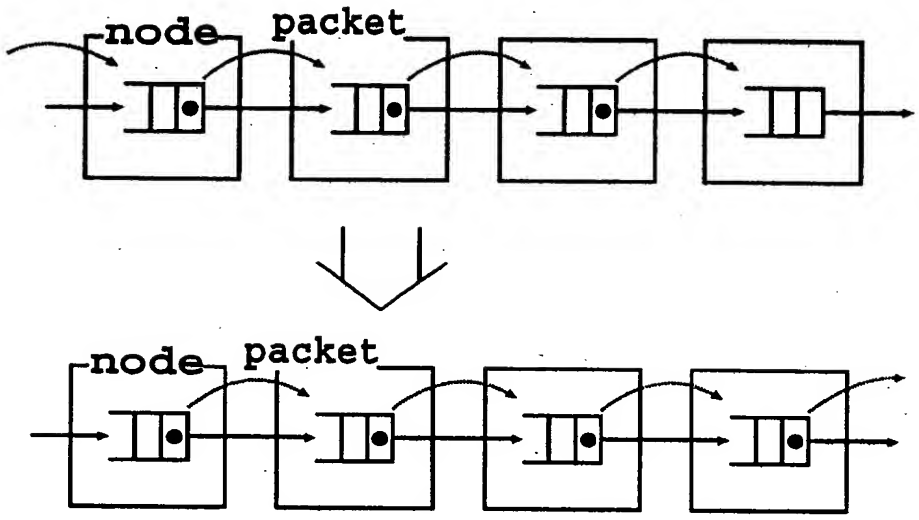


Fig.2

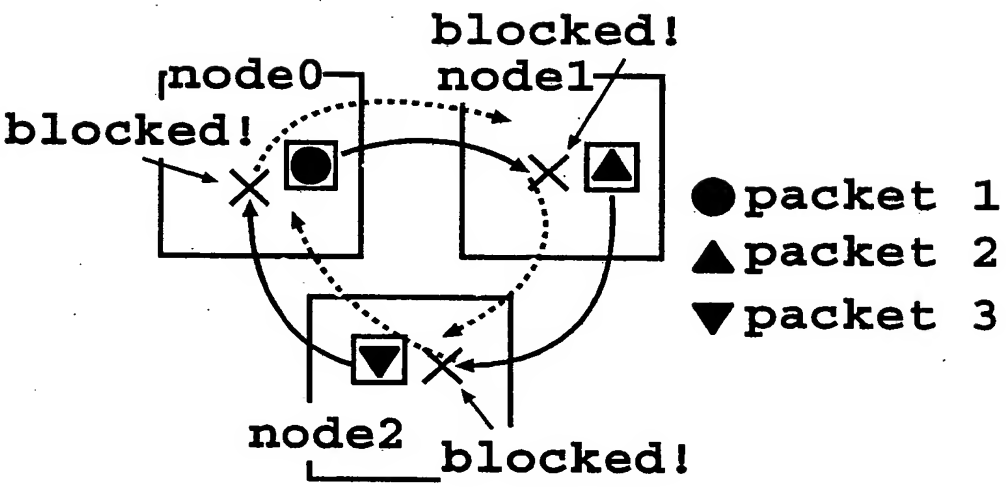

blocked!
node0
node1
blocked!
packet 1
packet 2
packet 3
node2
blocked!

Fig.3

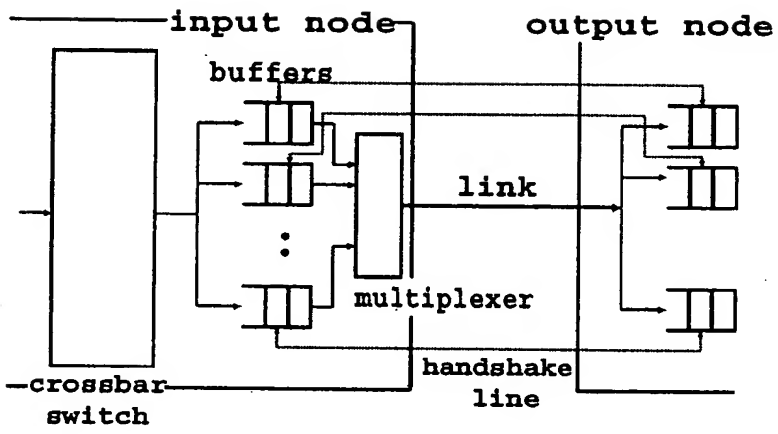

input node
output node
buffers
link
multiplexer
handshake line
crossbar switch

Fig. 4

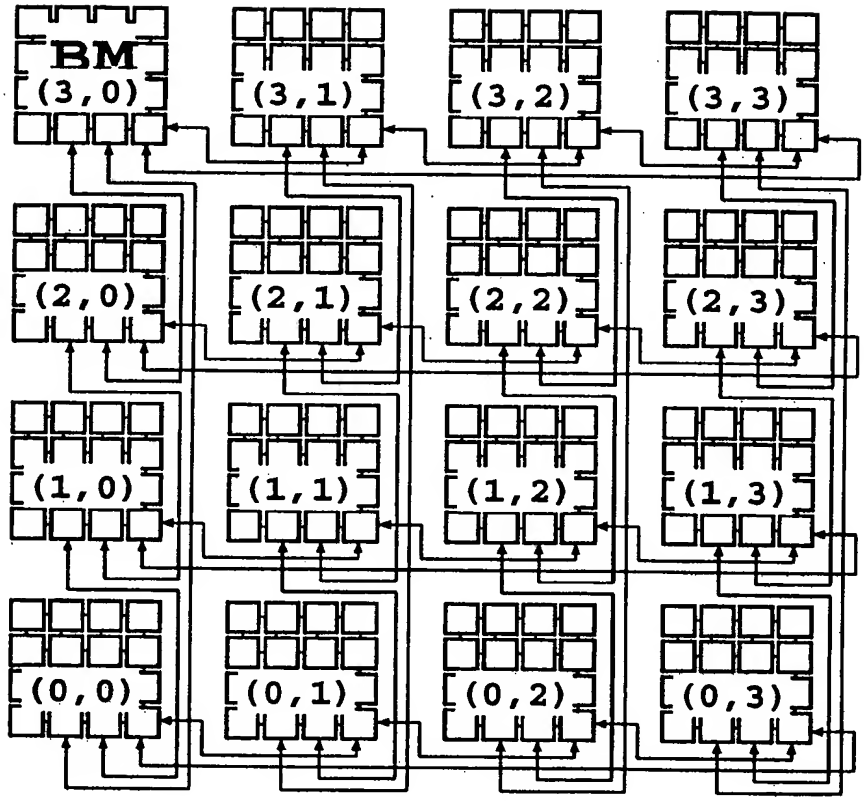


Fig.5

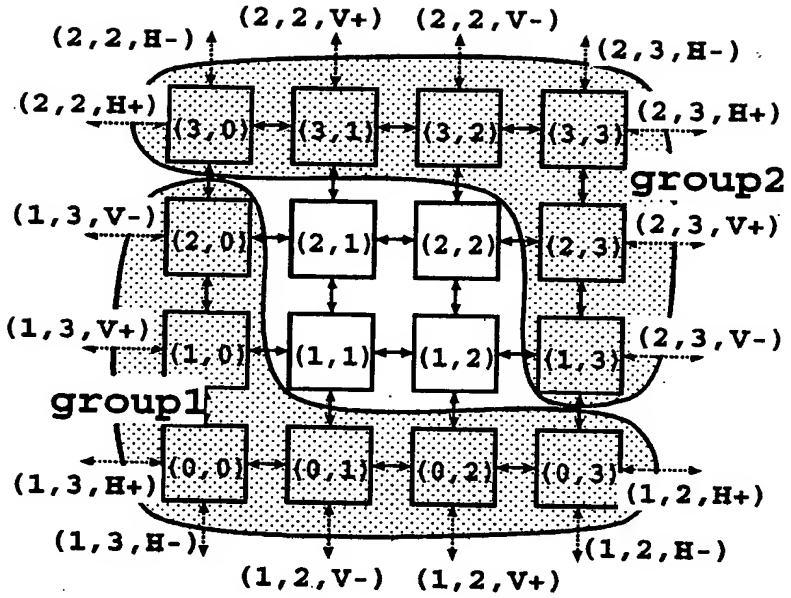

(2,2,V+)
(2,2,V-)
(2,2,H-)
(2,3,H-)
(2,2,H+)
(3,0)
(3,1)
(3,2)
(3,3)
(2,3,H+)
group2
(1,3,V-)
(2,0)
(2,1)
(2,2)
(2,3)
(2,3,V+)
(1,3,V+)
(1,0)
(1,1)
(1,2)
(1,3)
(2,3,V-)
group1
(0,0)
(0,1)
(0,2)
(0,3)
(1,3,H+)
(1,2,H+)
(1,3,H-)
(1,2,H-)
(1,2,V-)
(1,2,V+)

Fig. 6

```
Routing Algorithm for a Level-L TESH:

Routing(s,d);
source;s={s2L-1, s2L-2, ..., s0};  destination;d={d2L-1, d2L-2, ..., d0};
tag;t2L-1, t2L-2, ..., t0;      group;g ;

  for i = 2L-1:2;
    if (di-si+2^m) mod 2^m <= 2^m/2 then
            routedir = plus; ti= (di-si+2^m) mod 2^m;
    else  routedir = minus; ti= 2^m- (di-si+2^m) mod 2^m; endif;

    g = get_group_number(s,d,routedir);

    while(ti != 0) do
      if i is even number then
              outlet_nodex= outlet_x(g,i/2+1,H,routedir);
              outlet_nodey= outlet_y(g,i/2+1,H,routedir);endif;
      if i is odd number then                                        }(a)
              outlet_nodex= outlet_x(g,i/2+1,V,routedir);
              outlet_nodey= outlet_y(g,i/2+1,V,routedir);endif;
      BM_routing(outlet_nodex, outlet_nodey);

      if routedir = plus then send packet to next BM;
      else             send packet to previous BM; endif;          }(b)

      ti= ti- 1;
    endwhile;
  endfor;

    BM_routing(d1,d0);  ------------------------------------------> (c)
end.


BM_routing(dx, dy);
source;sx,sy; destination;dx,dy;
tag;tx,ty;

  tx= dx- sx;
  ty= dy- sy;
  while(ty != 0) do
    if ty > 0 then move packet to upper node; ty= ty- 1; endif;
    if ty < 0 then move packet to lower node; ty= ty+ 1; endif;
  endwhile;
  while(tx != 0) do
    if tx> 0 move packet to right node; tx= tx- 1; endif;
    if tx< 0 move packet to left node; tx= tx+ 1; endif;
  endwhile;
end.
```

Fig. 7

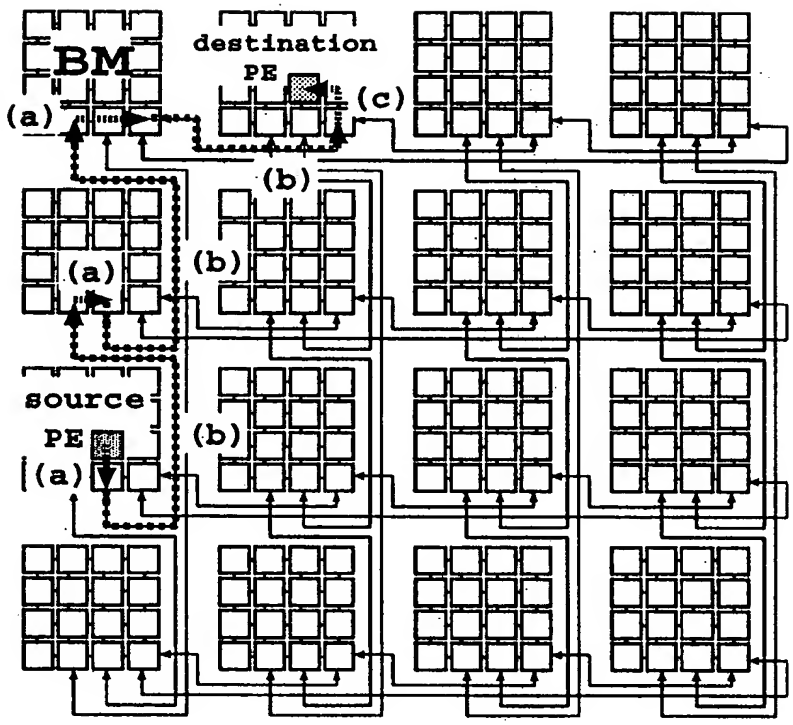


Fig. 8

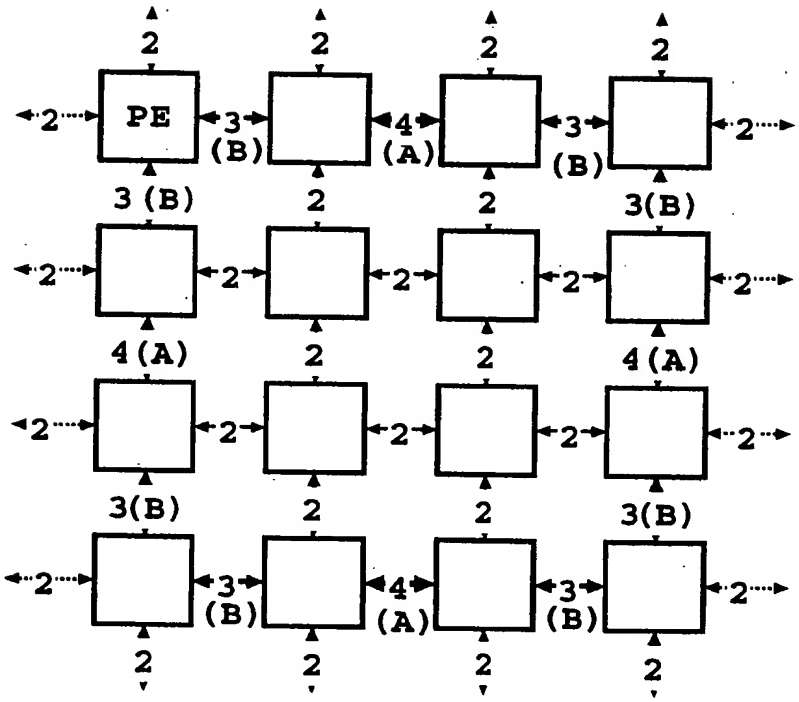


Fig.9

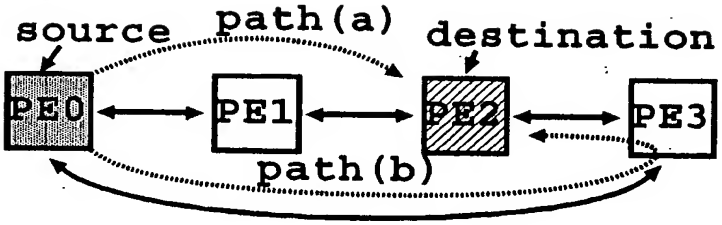


Fig.10

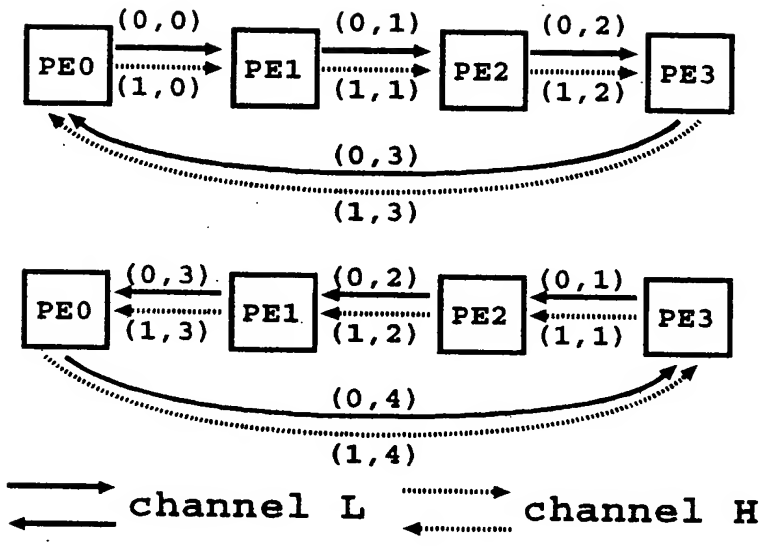
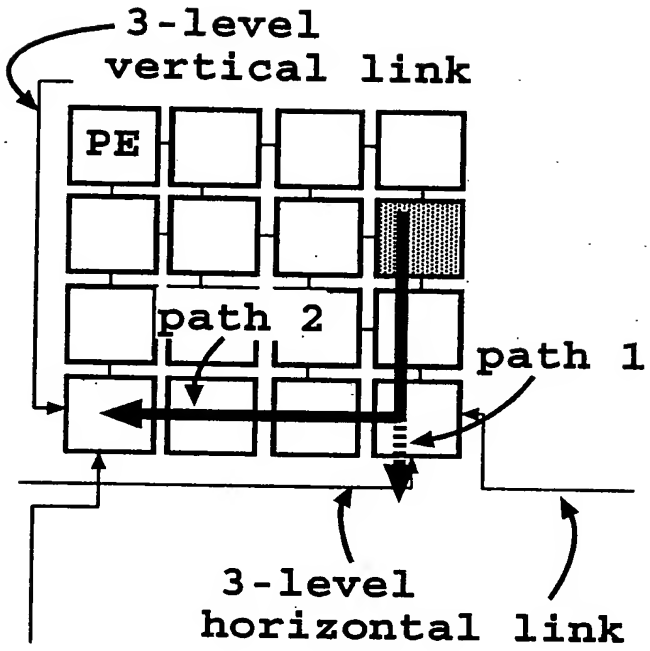

PE0
PE1
PE2
PE3
(0,0)
(1,0)
(0,1)
(1,1)
(0,2)
(1,2)
(0,3)
(1,3)
PE0
PE1
PE2
PE3
(0,3)
(1,3)
(0,2)
(1,2)
(0,1)
(1,1)
(0,4)
(1,4)
channel L
channel H

Fig.11


3-level
vertical link
PE
path 2
path 1
3-level
horizontal link